AD-A118 371 AERODYNE RESEARCH INC BILLERICA MA F/G 20/6
OPTICAL ARRAY PROCESSOR.(U)
JUL 82 N00014-82-C-0302
UNCLASSIFIED ARI-2622-10 NL

1 of 1
ADA
118371

END
DATE
FILMED
9-82
DTIC

ARI-2622-10

12

# OPTICAL ARRAY PROCESSOR

FINAL REPORT

Prepared By:

AERODYNE RESEARCH, INC.
*The Research Center At Manning Park*
45 Manning Road, Billerica, MA. 01821

Prepared For

Scientific Officer, Information Sciences
(Dr. Alan Meyrowitz)
Office Of Naval Research
800 N. Quincy Street, Arlington, VA. 22217

Contract No. N00014-82-C-0302

JULY, 1982

## TABLE OF CONTENTS

# EXECUTIVE SUMMARY

Because of its potential advantages in speed, size, cost and power consumption over digital computers, the special purpose optical computer has received much attention in the recent literature. The prototypical special purpose optical computer comprises a noncoherent vector x matrix multiplier with the results of the multiplication being detected, operated upon electrically, and serving as a basis for new inputs. This is an iterative optical processor. In this report we attempt a total reevaluation and rethinking of such a processor.

We begin the reevaluation with the reevaluation of the iterative concept itself. This concept is taken directly from the weld of the digital computer. It regards the calculation as something which inherently takes place in well defined steps. We then raise the question as to whether one could not derive differential equivalents to the iterative processor. Such algorithms might cause the output to relax continuously to the desired result. We have found that such "relaxation algorithms" can be derived and that they exhibit the property predicted for them.

The next step was to derive fully parallel methods of implementing the relaxation algorithms. We have designed hardware to accomplish this in a very straightforward manner. The net result is that the speed with which answers are calculated now becomes totally independent of the size of the problem.

Knowing the speed with which the parallel relaxation algorithm can be implemented electronically, we know the speed with which the processor must operate. That is, it does no good for the vector x matrix multiplier to operate any faster than the electronics that will be used to drive it. This led us to the question of whether or not we could make the analog vector x matrix multiplier itself electronic. An all electronic system would obviate a number of major difficulties the optical systems have encountered. In particular, the nonlinearities of source and detector are

irrelevant, real numbers are readily handled directly, and relatively high accuracy is readily achievable. We then replaced each component of the analog optical vector x matrix multiplier by its equivalent electrical counterpart. The resulting system is easy to build and achieves all of the desired features of the optical equivalent. In addition, it allows us easy electronic control of the matrix components. This, in turn, allows us to use the system in a variety of interesting ways which will be illustrated.

# 1.0 INTRODUCTION

The introduction of noncoherent vector x matrix optical operators[1,2] has led to the development of an assortment of iterative operations for the rapid performance of such problems as matrix inversion and eigenvector extraction[3,4]. In this contract Aerodyne Research, Inc. has reexamined all of these concepts and integrated them in a new and surprising way. The new problem solving system is simpler, cheaper, more flexible, and more accurate than its predecessor.

## 2.0 REVIEW OF PRIOR HARDWARE

Most optical vector x matrix multipliers now being studied fall into either of two classes. The first class[1,2,3,4] uses an array of light emitting diodes (LED's) to represent the input signals. The light from each LED is spread uniformly across one row of a spatial mask which represents the matrix. A subsequent optical system collects the light column by column to produce the desired outputs. Figure 1 shows such a system schematically. There are a number of difficulties with this system such as source and detector uniformity, source and detector nonlinearity, imperfections in the optical system, and the fact that the mask can only be changed conveniently if it is in the form of a rather expensive spatial light modulator. Recently a second approach has been described[5]. That approach, called "Optical Systolic Array Processing", involves the use of an acousto-optic modulator to input the vector information and a one dimensional array of light emitting diodes to input the matrix components. This system obviates a need for a spatial light modulator but it is very expensive and still at a very early stage of development. Thus we appear to have a choice between a mask on an expensive spatial light modulator or no mask and an expensive acousto-optic system.

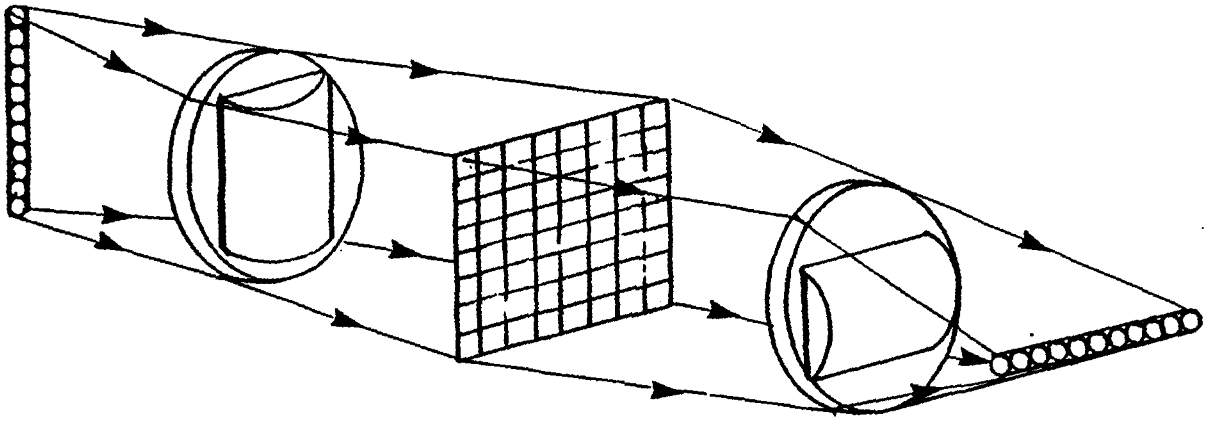


Figure 1. The prototypical optical vector x matrix multiplier comprises a column of light sources on the left, anamorphic lenses, a mask representing the matrix, further anamorphic lenses, and a row of detectors which give the output vector components.

## 3.0 REVIEW OF PRIOR FEEDBACK METHODS

While a number of authors have discussed feedback methods to implement iterative optical processors, only one appears to have given details of the implementation(2). In this system a micro computer is used to operate on the output vector to provide a new input vector. Of course, such a system immediately throws away any advantage that may have come from the parallel nature of the vector x matrix operation. Iterative operations for matrix inversion and solution of linear equations(4) and for eigenvector/eigenvalue solution(3) have been taken from the digital community and applied to these analog processors.

# 4.0 NEW FEEDBACK METHODS

## 4.1 Introduction

In our reanalysis of the system we will invert the historical development. That is, we will begin with the derivation of algorithms suitable for analog processors and then determine what sort of analog vector x matrix systems are required to utilize those algorithms well. Finally, we will develop the simplest hardware required to accomplish the desired operations.

Optical operations performable by properly-connected optical matrix x vector operators are numerous[1,2,3,4]. These operators perform the operation for the N x N matrix A,

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} = A \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \tag{1}$$

or, equivalently,

$$\underline{y} = A\underline{x} \tag{2}$$

in parallel. That is all the N components of $\underline{y}$ are calculated simultaneously. Typical use of these operators involves iterative operations[3,4] wherein

(1) an input vector $\underline{x}^{(o)}$ is chosen,

(2) $\underline{y}^{(o)} = A\underline{x}^{(o)}$ is calculated,

(3) some operation is performed on $\underline{y}^{(o)}$ to calculate a new vector $\underline{x}^{(1)}$,

(4) $\underline{y}^{(1)} = A\underline{x}^{(1)}$ is calculated, and the process is repeated until $\underline{y}^{(k+1)}$ and $\underline{y}^{(k)}$ become essentially equal.

These iterative algorithms are borrowed from digital computer domain. Implementation of these algorithms often use a hybrid combination of the optical matrix multiplier and a digital computer which controls the data path and steps through the iteration. Furthermore, most applications of those algorithms have not been fully parallel. Sequential treatments of the parallel data from the optical operators are processed by the digital computer in the loop. Accordingly, our goals have been twofold:

(1) design fully parallel algorithms and hardware,

(2) design analog feedback (relaxation) implementations rather than the iterative procedures.

These fully-parallel, relaxation algorithms should converge to the proper answer at a speed which is independent of the vector length N. What governs the convergence speed is (1) the inherent hardware speed and (2) the problem details.

### 4.2 Solution To Simultaneous Equations

In this example we shall consider the problem of solving for the vector $\underline{x}$ from the equation

$$A\underline{x} = \underline{y} \tag{3}$$

where $\underline{y}$ is a given real N x 1 vector. The N x N real matrix A is assumed to be of full rank and may have eigenvalues which are in general complex and may not be distinct. For purpose of illustration, let A be 2 x 2. The implementation can easily be generalized to the case of an N x N matrix. Consider first the circuit implementation as shown in Figure 2. The vector $\underline{y}$ is represented by the voltage sources $y_1$ and $y_2$. The construction of the circuit guarantees that if a stable equilibrium is attended, the resulting voltages $x_1$ and $x_2$ will be the solution. Nevertheless, it will be shown that the stability of the circuit depends on the eigenvalues of the matrix A.

The stability of the circuit may be analyzed by looking at its asymptotic behavior for an arbitrary starting condition. For this case, it suffices to

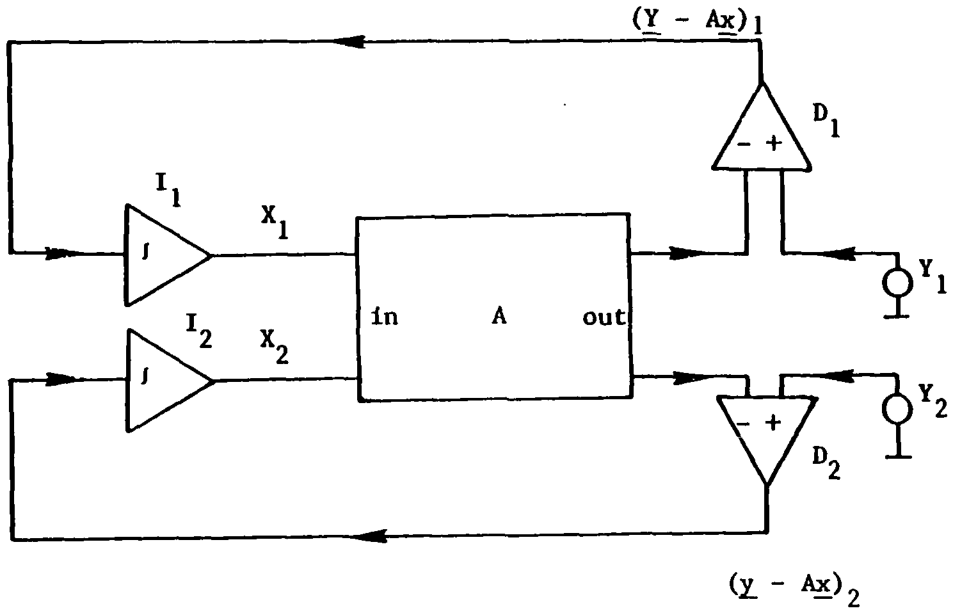


Figure 2. A conditionally stable matrix equation solver for a 2 x 2 matrix.

assume distinct eigenvalues for A. The circuit is described by the differential equation

$$\frac{d}{dt}\underline{x} = -\frac{1}{\tau}(A\underline{x} - \underline{y}) \tag{4}$$

where $\tau$ is the time response of the integrators $I_1$ and $I_2$. We may set $\tau$ equal to 1 by measuring t in units of $\tau$. The solution to (4), for an initial condition $\underline{x}_o$, is

$$\underline{x}(t) = \sum_{i=1}^{N} \left[ \underline{x}_o \cdot \underline{e}_i^{-\lambda_i t} + \frac{\underline{y}.\underline{e}_i}{\lambda_i}(1-e^{-\lambda_i t}) \right] \underline{e}_i \tag{5}$$

where the $\underline{e}_i$ are the eigenvectors with $\lambda_i$ the corresponding eigenvalues. If all the eigenvalues $\lambda_i$ are real and positive, the solution of (5) indeed converges to

$$\underline{x}(t \to \infty) = \sum_{i=1}^{N} \left(\frac{\underline{y} \cdot \underline{e}_i}{\lambda_i}\right) \underline{e}_i$$

$$= A^{-1}\,\underline{y} \tag{6}$$

If the real parts of $\lambda_i$ are all negative, the signs of the difference amplifiers $D_1$ and $D_2$ may be reversed and the correct asymptotic solution may still be obtained. If there are one or more of the real part of the eigenvalues of opposite signs, however, the solutions always diverge. This divergence will cause saturation of the amplifiers.

To guarantee stability of the system, the feedback system of Figure 2 is modified to implement the following transfer function:

$$\frac{d}{dt}\underline{x} = -\frac{1}{\tau}A^T(A\underline{x} - \underline{y}) \tag{7}$$

Here the superscript T indicates the matrix transpose. The corresponding circuitry is shown in Figure 3. Since $A^TA$ is a real and symmetric matrix

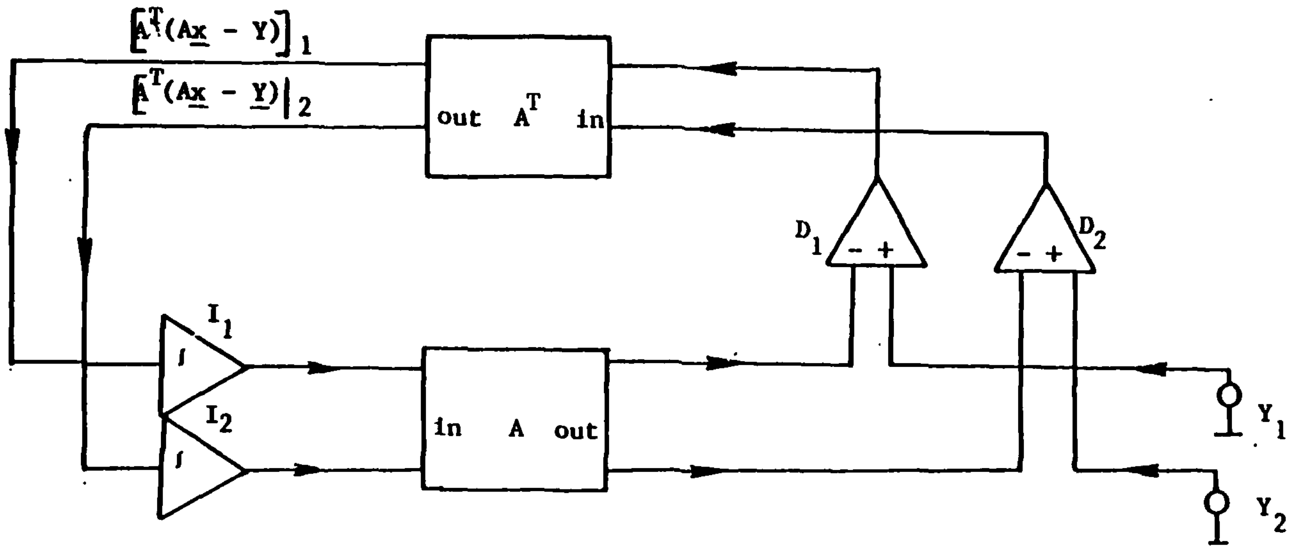

$[A^T(A\underline{x} - Y)]_1$
$[A^T(A\underline{x} - \underline{Y})]_2$
out
$A^T$
in
$D_1$
- +
$D_2$
- +
$I_1$
$I_2$
∫
∫
in
A
out
$Y_1$
$Y_2$

Figure 3. A stable matrix equation solver.

all its eigenvalues $\lambda_i$ are real and positive, and there exists a real orthogonal matrix S such that

$$S\,A^T\,A\,S^T = D \tag{8}$$

where

$$D = \begin{bmatrix} \lambda_i & & 0 \\ & \ddots & \\ 0 & & \lambda_N \end{bmatrix} \tag{9}$$

The solution to (7) for an arbitrary initial condition $\underline{x}_o$ is then

$$\underline{x} = S^T \begin{bmatrix} e^{-\lambda_i t} & & 0 \\ & \ddots & \\ 0 & & e^{-\lambda_n t} \end{bmatrix} S\underline{x}_o + S^T \begin{bmatrix} \frac{1 - e^{-\lambda_i t}}{\lambda_i} & & 0 \\ & \ddots & \\ 0 & & \frac{1-e^{-\lambda_n t}}{\lambda_i} \end{bmatrix} SA^T\underline{y} \tag{10}$$

The positive definiteness of $\lambda_i$ guarantees stability of the system and the asymptotic solution of $\underline{x}$ is

$$\underline{x}(t \rightarrow \infty) = S^T \begin{bmatrix} \frac{1}{\lambda_1} & & 0 \\ & \ddots & \\ 0 & & \frac{1}{\lambda_n} \end{bmatrix} S\,A^T\underline{y} \tag{11}$$

From (8), postmultiplying by $(AS^T)^{-1}$, we obtain

$$SA^T = DS\,A^{-1} \tag{12}$$

Combining (11) and (12) we obtain the desired result

$$\underline{x} = A^{-1}\,\underline{y} \tag{13}$$

The convergence of (8) from an arbitrary initial condition $\underline{x}_o$ is therefore depending on the minimum value $\lambda_m$ of the set of $\lambda_i$ and goes as $e^{-\lambda_m t}$ regardless of the size of the matrix. This is a result of the fully parallel nature of the implementation.

The above argument can be applied to a complex matrix A and a complex vectory $\underline{y}$. Then complex number multipliers have to be used and the transpose matrices $A^T$, $S^T$ will be ɾeplaced by the hermitian ajoints $A^\dagger$ and $S^\dagger$. The matrix S will then be arbitrary.

When the matrix A is not of full rank, the matrix $A^TA$ will be positive semi-definite. The algorithm will relax to a vector $\underline{x}$ which satisfies (3). Since $\underline{x}$ is not unique, the particular asymptotic value for $\underline{x}$ will be dependent on the initial condition $\underline{x}_o$.

### 4.3 Solution Of Eigenvector/Eigenvalue Problems

We specialize again (without loss of generality) to the 2 x 2 case. If the equation

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = (G) \left[ A \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \right] , \qquad (14)$$

can be satisfied, then the vector

$$\vec{e} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \qquad (15)$$

is an eigenvector and the gain

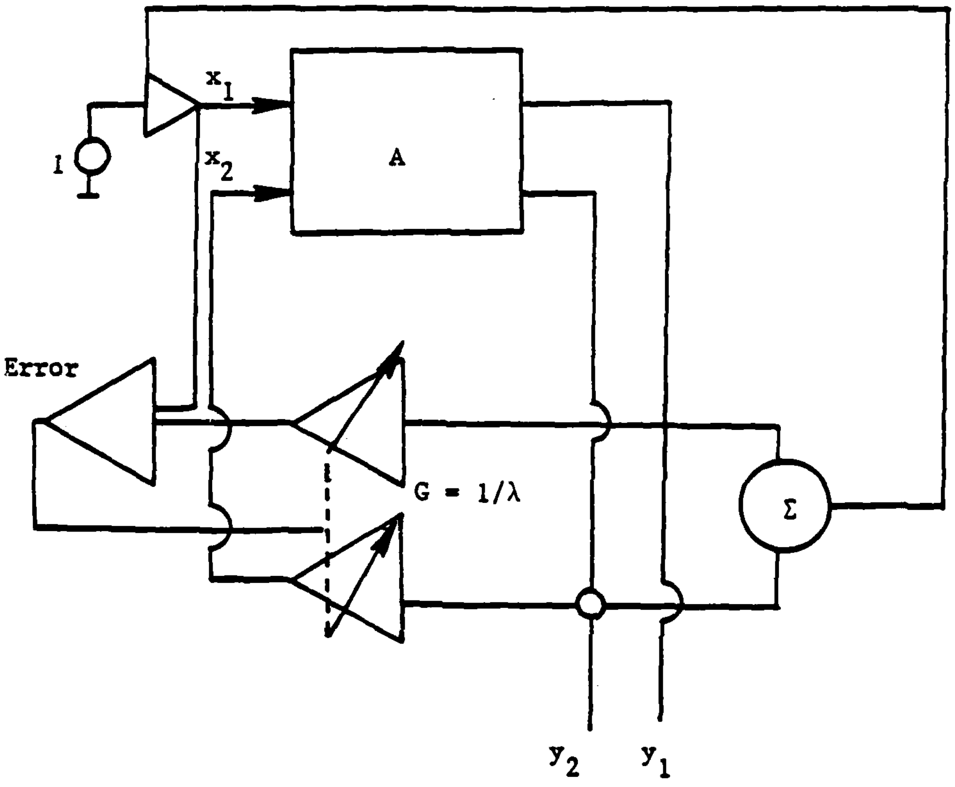
$$G = 1/\lambda \qquad (16)$$

is the reciprocal of the corresponding eigenvalue.

Figure 4 shows a system designed to vary the gain in such a way that the eigenvalue/eigenvector problem is solved. An $x_1$ is input to the system to generate $y_1$ and $y_2$. The difference between $x_1$ and $y_1$ drives the gain from both the $y_1$ and $y_2$ channels in such a way that the system is driven to an eigenvector. The same normalization problems can occur in the differential system as occur in the discrete implementation of the power method of finding eigenvectors. In that method, normalization is achieved after each step by driving the total "energy" in each iteration of the vector to unity. The equivalent operation here is the driving of the gain on the input to the system from the sum of $y_1$ and $y_2$. This guarantees that the energy in the eigenvector cannot become either too low or too high. It is eady to see that this approach is the differential analog of the power method. Thus the first eigenvalue found will be the eivenvalue which has the highest absolute value.

## 4.4 Discussion

The relaxation algorithm we have described is clearly the differential version of the iterative algorithm[2,3]. Our "algorithm" for generating these relaxation algorithms is to drive the circuit in such a way as to embody the definition of the solution by driving the inputs using the error signals. In this case we drive $\underline{x}$ using $A\underline{x} - \underline{y}$. There is no definite number of iterations. Rather $A\underline{x} - \underline{y}$ converges smoothly (not in discrete jumps) to zero. This approach appears to take full advantage of the analog nature of the optical processor.



$$\binom{y_1}{y_2} = (\frac{1}{\lambda}) \left[ A\binom{y_1}{y_2} \right]$$

| | | |
|---|---|---|
| Found | Gain<br>Found | Given |

**Figure** **A preliminary design for a relaxation parallel system to calculate eigenvalues and eigenvectors.**

## 5.0 NEW VECTOR TIMES MATRIX MULTIPLIER HARDWARE

Now that we possess algorithms specific to and appropriate for an analog vector x matrix operator, we can determine the requirements for that operator to utilize the algorithm well. Because the algorithm and the hardware to implement it are inherently parallel, we will wish to use parallel vector x matrix multiplication. This means that we will want to use the earlier type system which utilized a mask. If would be convenient if the mask were both inexpensive and readily changeable. As well, it would be convenient if we were able to handle real numbers as opposed to the present noncoherent optical systems which can handle only non-negative numbers. All of this suggests that perhaps optics is not the optimum approach to this particular analog processor. A total revision of thinking was brought about in a way best described by Baum in The Wizard Of Oz:

> Thereupon the Wicked Witch enchanted my ax, and when I was chopping away at my best one day, for I was anxious to get the new house and my wife as soon as possible, the ax slipped all at once and cut off my left leg.
>
> This at first seemed a great misfortune, for I knew a one-legged man could not do very well as a woodchopper. So I went to a tinsmith and had him make me a new leg out of tin. The leg worked very well, once I was used to it. But my action angered the Wicked Witch of the East, for she had promised the old woman I should not marry the pretty Munchkin girl. When I began chopping again, my ax slipped and cut off my right leg. Again I went to the tinner, and again he made me a leg out of tin. After this the enchanted ax cut off my arms, one after the other; but, nothing daunted, I had them replaced with tin ones. The Wicked Witch then made the ax slip and cut off my head, and at first I thought that was the end of me. But the tinner happened to come along, and he made me a new head out of tin.

One by one we have replaced all of the electro optic components by strictly electrical ones. In each case, we have gained in speed and sensitivity. The input signal is broadcast to an array of variable gain amplifiers which represent the matrix components. Because we are working in the electrical rather than the optical domain the handling of negative as well as positive components becomes quite easy. The appropriate signals from the variable gain amplifiers are summed to produce the required outputs. A block diagram of a 3 x 3 operator is shown in Figure 5. Figure 6 shows an actual implementation of this system. Figure 7 gives a detailed schematic diagram of the components used in making the system. These components were chosen for convenience rather than for their optimality for this purpose. Nevertheless, the accuracy we achieve is far greater than that ever achieved by any of the prior optical systems.

The primary disadvantage of the all-electrical processor relative to its opto-electric counterpart is that there is a potential loss of speed. Whether that potential loss is real seems quite questionable since the controls of the optical system are themselves electrical and operate at the same speed as our entire electrical system. In addition, the relaxation algorithm we derived is implemented in electronic form so that it itself would limit the speed even if it operated on an inherently-faster optical vector x matrix operator.

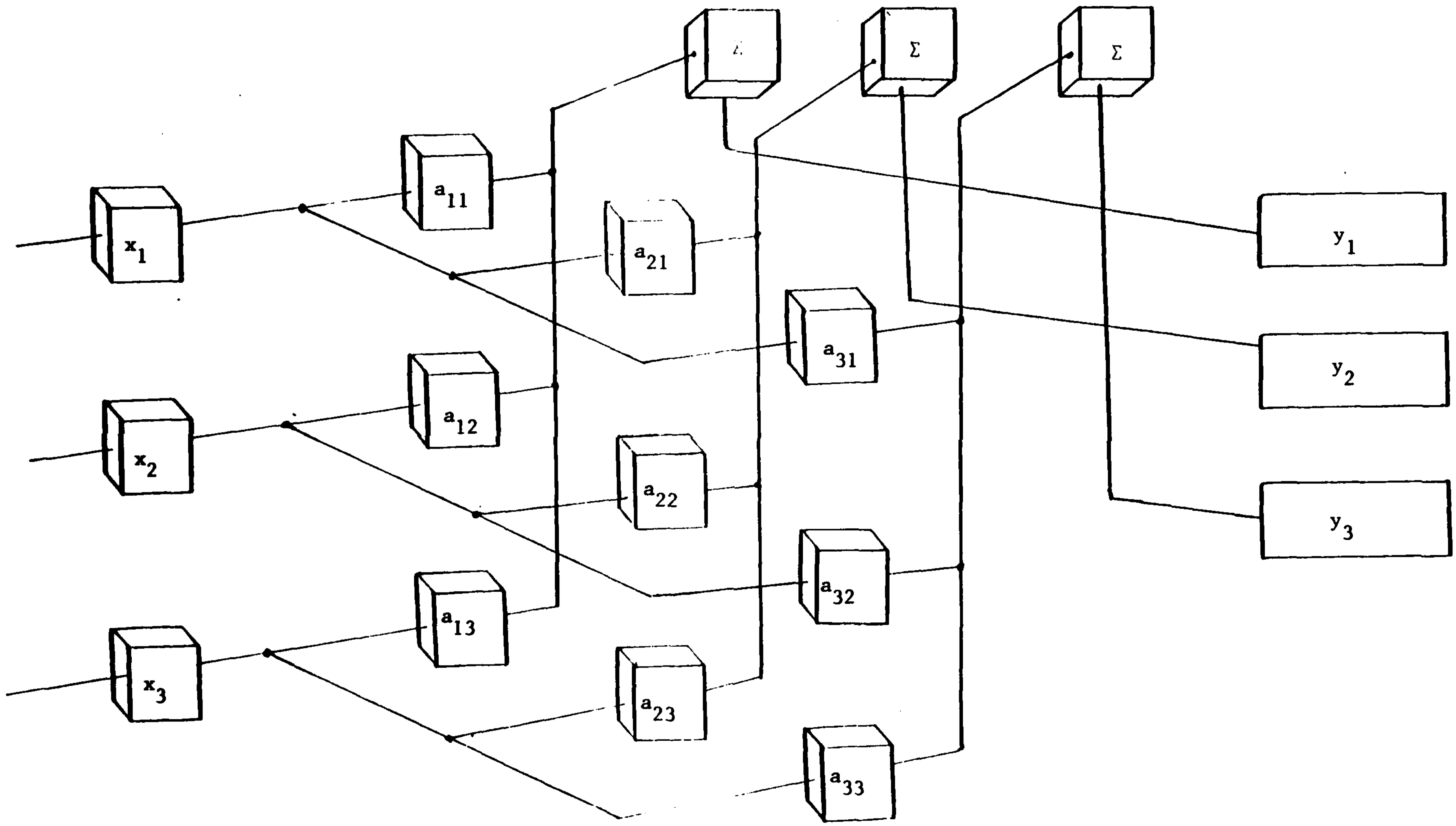


Figure 5. Electrical voltages (positive of negative) represent the input vector components $x_1$, $x_2$ and $x_3$. Variable gain amplifiers (positive or negative) represent the matrix components. Appropriate amplified signals are summed to produce the output components $y_1$, $y_2$ and $y_3$ which can also be positive or negative.

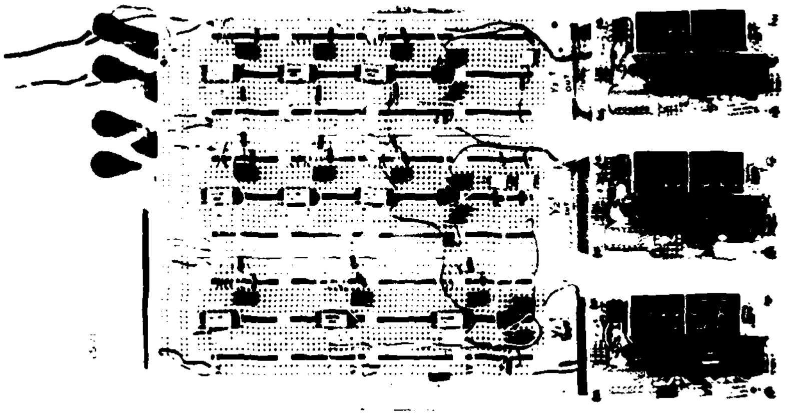


Figure 6. Shown here is a photograph of a 3 x 3 vector x matrix multiplier. The output signals are displayed on the digital displays to the right.

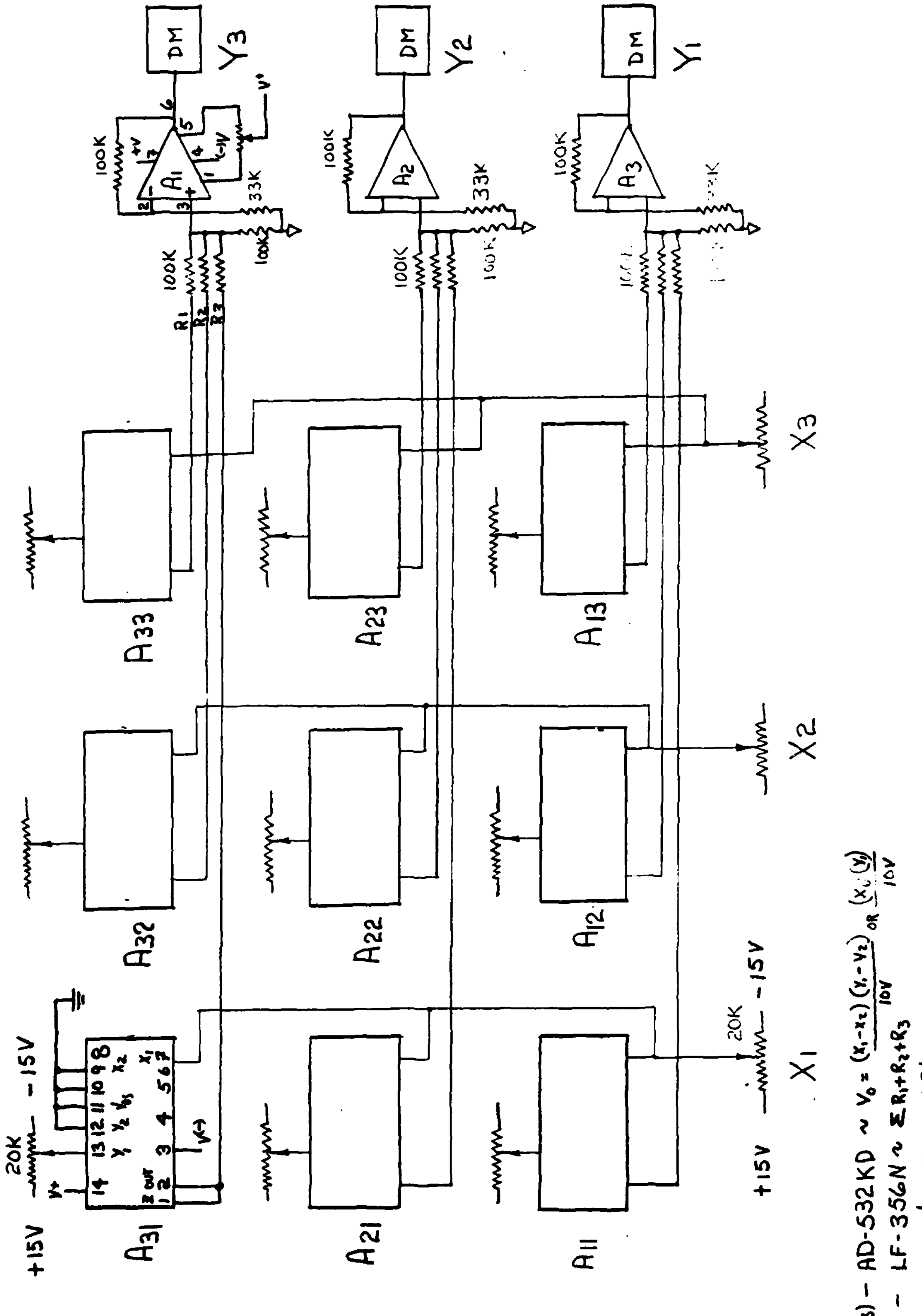


Figure 7. Shown here is a more detailed block diagram indicating components used in our implementation.

## 6.0 TEST RESULTS

The 2% accurate components operated to better than 2% over their rated ranges and (except for a drift which can be detected and nulled out) all calculations were good to 2% or better. Of course, the repeatability of errors is great, so by selecting better components we can readily drop the errors to less than 1%. Positive and negative numbers were easily handled. This compares with optics which requires great machinations(1,2,3,4) to deal with these at all.

## 7.0 SYSTOLIC INTERCONNECTIONS

These basic units can be interconnected to form systolic-type arrays. This type of configuration is made possible by the fact that, unlike its optical counterpart, this analog vector x matrix multiplier has readily addressable matrix component information. The following drawings show the configuration of a single local processing unit in schematic form (Figure 8), the use of multiple local processing units in a systolic array processor (Figure 9), and the use of a single local processing unit in a configuration somewhat like systolic array processing (Figure 10). These systolic arrangements allow us to use local processing units capable of handling N x N matrices in operations on much larger matrices.

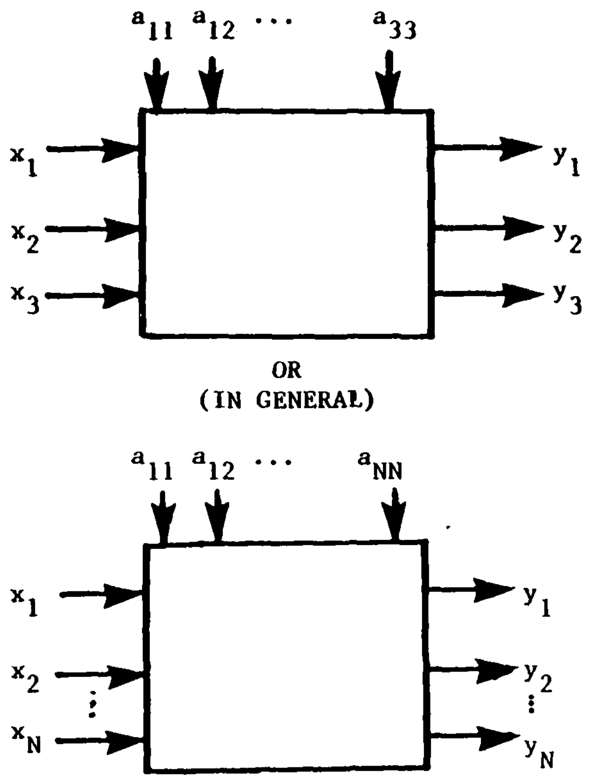
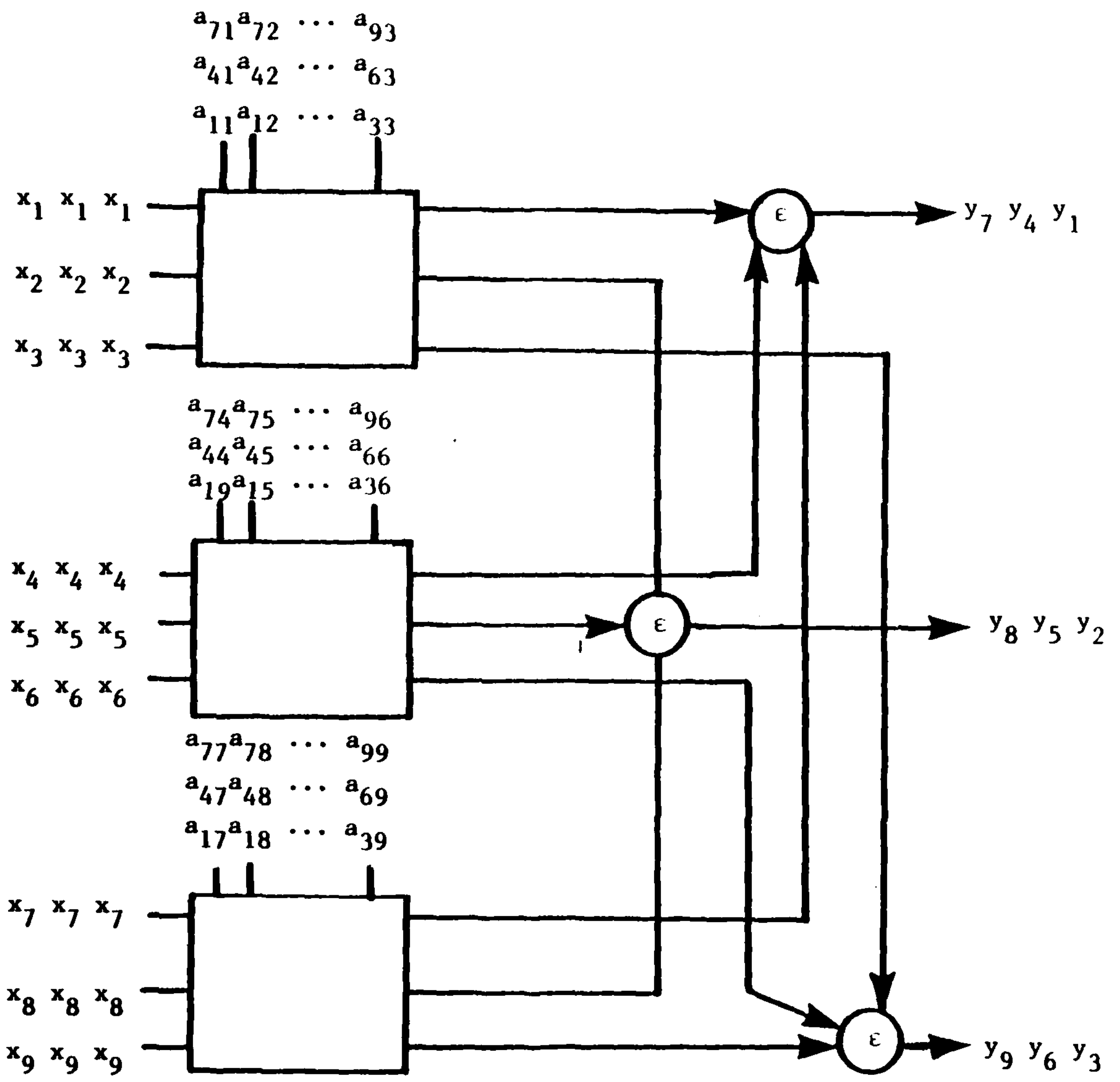
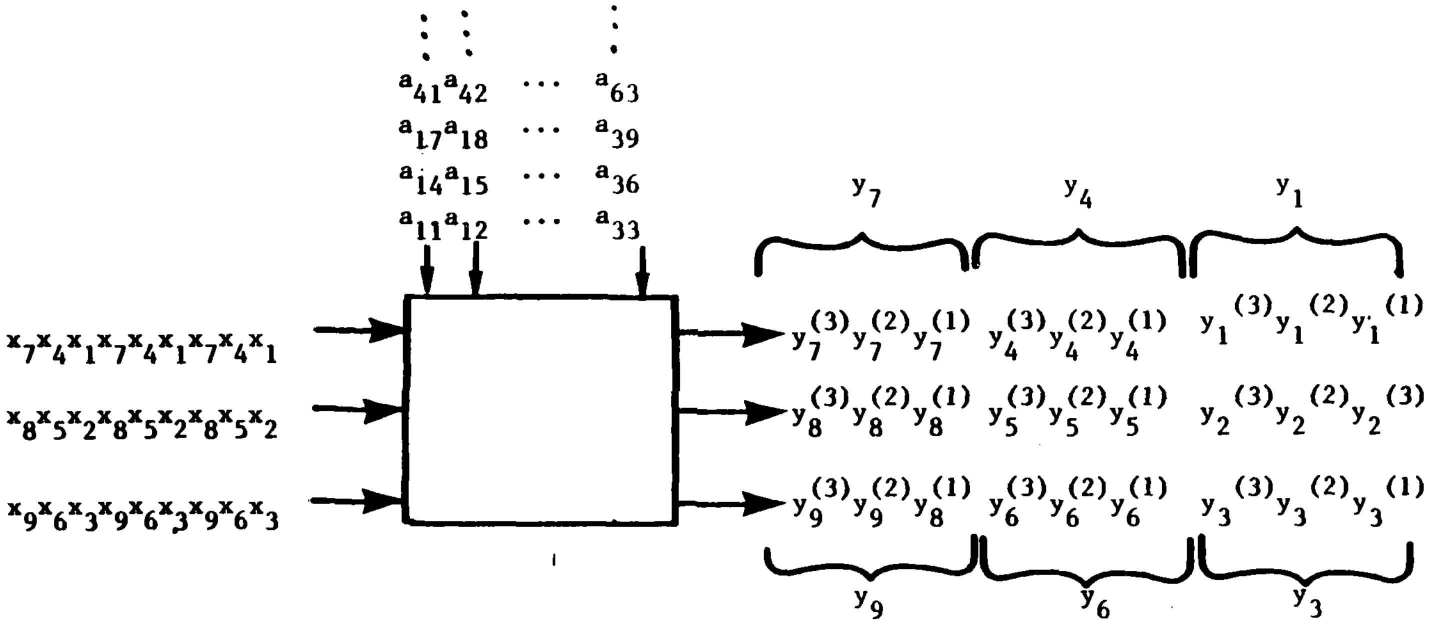


Figure 8. The local processing unit devised under this contract can be represented symbolically as indicated here. This symbolism is useful in showing how these local processing units can be connected.



$a_{71}a_{72} \cdots a_{93}$
$a_{41}a_{42} \cdots a_{63}$
$a_{11}a_{12} \cdots a_{33}$
$x_1$ $x_1$ $x_1$
$x_2$ $x_2$ $x_2$
$x_3$ $x_3$ $x_3$
$a_{74}a_{75} \cdots a_{96}$
$a_{44}a_{45} \cdots a_{66}$
$a_{19}a_{15} \cdots a_{36}$
$x_4$ $x_4$ $x_4$
$x_5$ $x_5$ $x_5$
$x_6$ $x_6$ $x_6$
$a_{77}a_{78} \cdots a_{99}$
$a_{47}a_{48} \cdots a_{69}$
$a_{17}a_{18} \cdots a_{39}$
$x_7$ $x_7$ $x_7$
$x_8$ $x_8$ $x_8$
$x_9$ $x_9$ $x_9$
$\varepsilon$
$\varepsilon$
$\varepsilon$
$y_7$ $y_4$ $y_1$
$y_8$ $y_5$ $y_2$
$y_9$ $y_6$ $y_3$

Figure 9. Multiple local processing units can be used to perform matrix x vector operations on extremely large matrices as indicated here.



$a_{41}a_{42} \cdots a_{63}$
$a_{17}a_{18} \cdots a_{39}$
$a_{14}a_{15} \cdots a_{36}$
$a_{11}a_{12} \cdots a_{33}$
$x_7x_4x_1x_7x_4x_1x_7x_4x_1$
$x_8x_5x_2x_8x_5x_2x_8x_5x_2$
$x_9x_6x_3x_9x_6x_3x_9x_6x_3$
$y_7$
$y_4$
$y_1$
$y_7^{(3)}y_7^{(2)}y_7^{(1)}$ $y_4^{(3)}y_4^{(2)}y_4^{(1)}$ $y_1^{(3)}y_1^{(2)}y_1^{(1)}$
$y_8^{(3)}y_8^{(2)}y_8^{(1)}$ $y_5^{(3)}y_5^{(2)}y_5^{(1)}$ $y_2^{(3)}y_2^{(2)}y_2^{(3)}$
$y_9^{(3)}y_9^{(2)}y_8^{(1)}$ $y_6^{(3)}y_6^{(2)}y_6^{(1)}$ $y_3^{(3)}y_3^{(2)}y_3^{(1)}$
$y_9$
$y_6$
$y_3$

Figure 10. A single processing unit can be fed information in such a way that it itself generates the information required for the larger vector x matrix multiplier as shown here.

## 8.0 CONCLUSIONS

In taking these last few steps in the evolution of analog processing, we have repeated a process which is followed often in many fields of engineering. Historically, the hardware is invented first and appropriate software is designed around it. Upon reflection, we often find that the algorithms which we have borrowed from other fields are not as effective as those which we can derive specifically for this type of hardware. Having derived appropriate algorithms we often find that the hardware that stimulated this whole line of thought must itself be reexamined. Thus through the mechanism of continued research on optical processing we have developed a non-optical processor which appears to be as fast as needed for the fully-parallel processing contemplated, yet much simplier and potentially more accurate than its optical counterpart.

We believe that the speed and simplicity of this system when combined with the increased simplicity of the algorithm derived here, allow us to make analog array processors of extremely high speed which are fully parallel. These processors should exceed in speed and accuracy specifications any optical system could produce. The reason for this assertion is that the optical systems have all of the same components but in addition, have electrical-to-optical and optical to electric conversions as well. This increased speed and simplicity should allow this family of relaxation electrical array processors to meet many of the government's needs wich regard to compactness, power consumption, and speed.

## REFERENCES

1. J. W. Goodman, A. R. Dias, and L. M. Woody, Opt. Lett., 2, 1 (1978).

2. D. Psaltis, D. Casasent, and M. Carlotto, Opt. Lett., 4, 348 (1979).

3. H. J. Caulfield, D. Dvore, J. W. Goodman, W. H. Rhodes, Appl. Opt., 20, 2263 (1981).

4. J. W. Goodman and M. S. Song, Appl. Opt. 21, 502 (1982).

5. H. J. Caulfield, et al., Optics Communications, Vol. 40, No. 2, (1981).